

Пермский филиал федерального государственного автономного  
образовательного учреждения высшего образования  
«Национальный исследовательский университет  
«Высшая школа экономики»

*Факультет экономики, менеджмента и бизнес-информатики*

Виноградов Никита Андреевич

## **ОЦЕНКА АРХИТЕКТУРЫ ПРОГРАММНОЙ СИСТЕМЫ ДЛЯ ЗАДАННОЙ ПРЕДМЕТНОЙ ОБЛАСТИ**

*Лабораторная работа*

студента образовательной программы «Программная инженерия»  
по направлению подготовки 09.03.04 Программная инженерия

Руководитель  
к.т.н., доцент кафедры Информа-  
ционных технологий в бизне-  
се НИУ ВШЭ-Пермь

---

А.В. Кычкин

Пермь, 2020 год

# Оглавление

<b>Глава 1. Оценка архитектуры программной системы для заданной предметной области.....</b>	<b>3</b>
1.1 Описание проекта . . . . .	3
1.2 Компоненты системы . . . . .	4
1.3 Архитектурный стиль . . . . .	5
<b>Глава 2. Диаграмма вариантов использования проектируемой системы в соответствии с нотацией UML .....</b>	<b>7</b>
<b>Глава 3. Архитектура информационной системы для заданной предметной области с использованием выбранного архитектурного решения.....</b>	<b>8</b>
3.1 Диаграммы активности и последовательности . . . . .	8

# Глава 1. Оценка архитектуры программной системы для заданной предметной области.

## 1.1. Описание проекта

*Описание проекта* - Автоматизированная система нагрузочного тестирования для веб-серверов (облачных систем).

*Цель проекта* - Автоматизировать и упростить процесс тестирования нагрузки и последующего исследования показателей для принятия решений.

*Задачи проекта* -

*Области применения* - Облачные распределенные системы, ETL и MapReduce системы.

*Составные части:*

1. Сервер системы - выполняет роль единой точки обращения к системе, позволяет назначать работы для воркеров и управляет работой Воркеров.
2. Воркеры - рабочие элементы системы выполняющие роль тестировщиков, в данной системе они могут выполнять 2 вида работ:
  - a) Hit-Based тестирование.
  - b) Сценарное тестирование.
3. Мониторинг - используется для просмотра информации о проведенных нагрузочных тестах, составлении графиков зависимостей.
4. Сборщик логов - используется для сбора информации со всех элементов системы и записи процесса выполнения на диск, для последующей отладки или просмотра.
5. Агенты - помощники основной системы, в основной части выполняются на target машине для которой выполняется тестирование чтобы отследить данные по процессору, памяти, дисковой нагрузке, также агенты используются для предотвращения падения системы в случае высокой нагрузки сервера нагрузочного тестирования.

6. Планировщик - используется для организации работ, которые выполняют воркеры. В текущей системе распределяет потоки выполнения для увеличения общей вычислительной мощности.
7. Веб-Сервер - используется для загрузки конфигурации нагрузочных тестов и просмотра информации по возможностям системы и свободным воркерам.
8. CLI-клиент - клиент системы выполняющий функции веб-сервера, только в режиме терминала.

## 1.2. Компоненты системы

1. Генератор запросов - используется самописный генератор Http/Https запросов на Golang и C++ со вставки на NASM, данное решение было принято после тестирования существующих генераторов и определения узких мест.
2. СУБД - для данной системы была выбрана колоночная Time-Series База данных ClickHouse. Данное решение обусловлено возможностями системы по предоставлению нагрузки и генерации метрик в ходе тестирования, (Было проведено тестирование возможностей системы и при режиме тестирования *максимальной производительности*, система может выдавать 1.8 Млн RPS) в ходе этого необходимо эффективно доставлять метрики в базу данных а не копить в памяти сервера.
3. Jenkins X Api Client - инструмент для подключения к системе сборки и доставки приложений.
4. Gitlab CI/CD Api Client - инструмент для подключения к системе сборки и доставки приложений.
5. Kubernetes Api Client - инструмент для подключения к системе оркестрации контейнеров, будет использовать для проверки масштабирования компонентов облачных систем.
6. Webhooks -инструменты создания конечных точек в системе на сервере, для выполнения определенных команд.

7. gRPC - инструмент эффективного подключения сервисов внутри и между дата-центрами с помощью подключаемой поддержки для балансировки нагрузки, трассировки, проверки работоспособности и аутентификации
8. TabiX - инструмент для анализа данных из ClickHouse и построения графиков и диаграмм по тестируемым системам

### **1.3. Архитектурный стиль**

1. Клиент-Сервер - используется для связи CLI Client и Воркеров в серверной системе.
2. Master-Slave - используется для масштабирования воркеров, в задачах тестирования серверов с большими возможностями.
3. Observer - используется для сбора логов и метрик в компонентах системы.

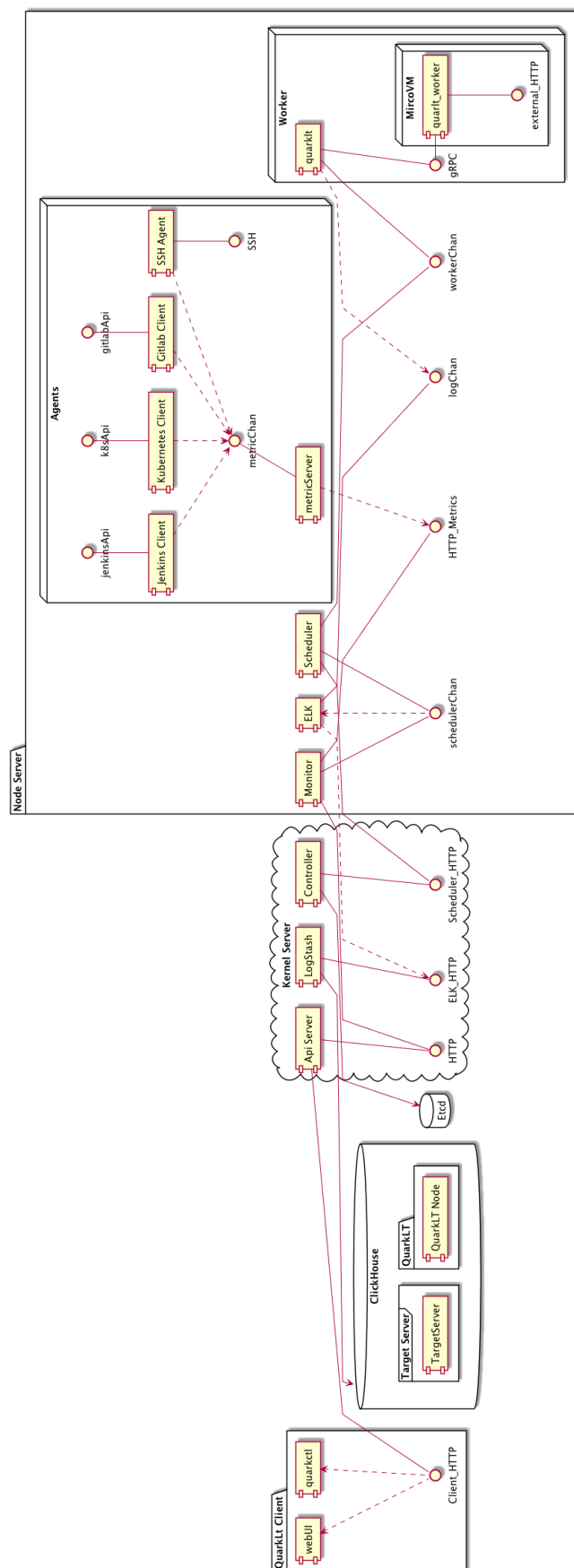


Рис. 1.1. Диаграмма компонентов системы системы

## Глава 2. Диаграмма вариантов использования проектируемой системы в соответствии с нотацией UML

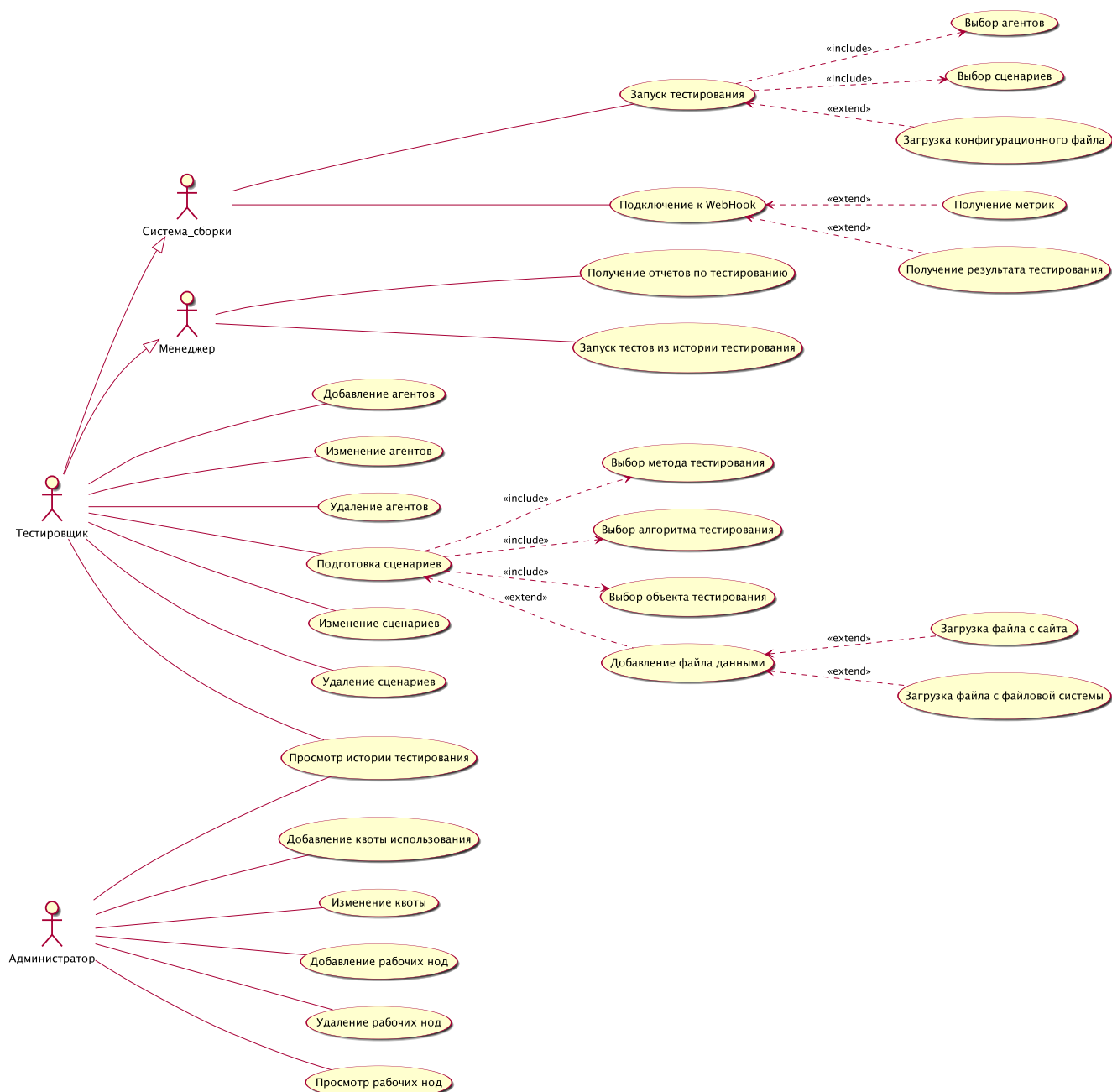


Рис. 2.1. Диаграмма прецедентов системы

# Глава 3. Архитектура информационной системы для заданной предметной области с использованием выбранного архитектурного решения

## 3.1. Диаграммы активности и последовательности

Диаграммы активности и последовательности описаны ниже

**Наименование:** Добавление рабочих нод.

**Обязанности:** Добавить рабочие ноды

**Ссылки:** прецедент «Добавление рабочих нод».

**Предусловия:** Администратор зашел в систему и выбрал «Добавление рабочих нод».

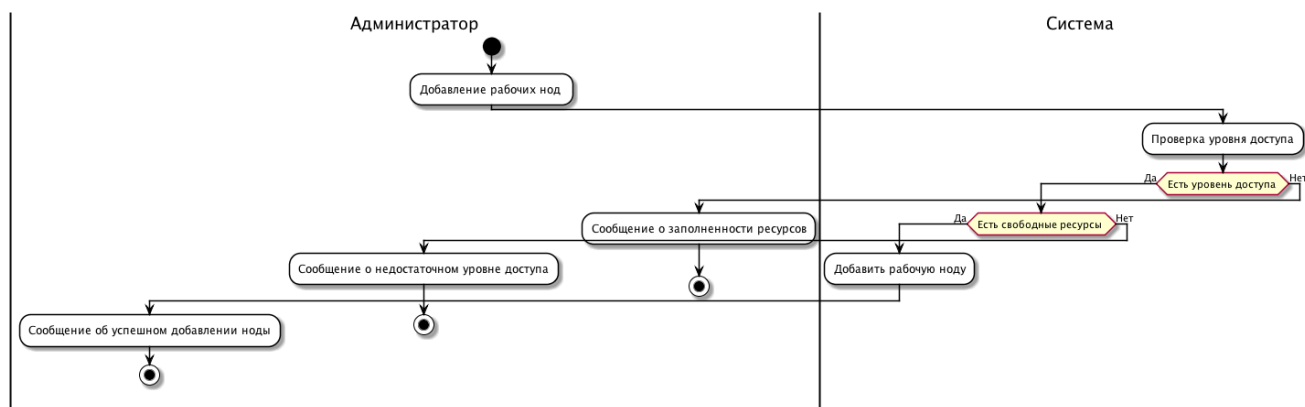


Рис. 3.1. Диаграмма последовательности